Catalog services for ATLAS

Version 0.10

David Adams
February 4, 2004

Introduction

The AJDL model [1] envisions that data is described with datasets and the provenance of that data with applications, tasks and jobs. These objects, their associations and their assigned metadata must be recorded and be made available to users and the services that act on their behalf. We assume the recording and retrieval of this information is done through services (presumably web services) that we collectively call catalog services.

This document presents a series of specific use cases that are used to identify the catalog services required for the upcoming ATLAS data challenge (DC2).

Catalog categories

The AJDL document identifies three categories of catalog services: repository, selection (or metadata) catalog and replica catalog. The interfaces for these services are described in that document. A repository is fully specified by the type of object it holds, e.g. a task repository holds task descriptions indexed by task ID. A selection catalog is effectively a relational table and is specified by its schema which includes an ID of the same type as the catalog, e.g. a task selection catalog has a unique task ID in each row. A replica catalog holds associations between a reference ID and a collection of replica ID's; e.g. a logical file and its physical replicas.

All objects (datasets, tasks …) are recorded in the corresponding repository when they are created or used. We generally do not explicitly indicate repository access in the use cases but note here that provenance tracking requires that any object whose ID appears in a selection or replica catalog must be recorded in a repository. We do identify interactions with other types of catalogs to illustrate their use and provide justification for their existence.

Virtual data catalog

In addition to the above, we assume the existence of a virtual data catalog (VDC) that tracks the provenance of virtual datasets. A minimal implementation would hold the application, task and parent dataset used to construct each cataloged dataset, or more precisely the ID's for each these entities. The parent dataset is replaced with a list of parents when datasets are merged. This information associated with each cataloged dataset is called the essential provenance; complete provenance would add history including details about splitting, when and where processing was done, etc.

The cataloged and parent datasets are virtual rather than concrete because we expect that users searching for data will not and should not be concerned with data history (beyond essential provenance), details of how and where data is placed, or even whether data yet exists. Instead we assume the existence of a data service that, on demand, finds the best

representation (concrete replica) of a virtual dataset, possibly creating the replica and its ancestors. The VDC provides the (abstract) prescriptions for creating datasets and, as such, serves as a catalog of the datasets (more precisely dataset prescriptions) that have been deemed interesting.

These abstract prescriptions are generally not sufficient for users to locate data. Instead, the producers of the data would like to attach metadata that reflects the full provenance chain as well as additional information. Thus we envision the existence of a separate dataset selection catalog (DSC) again for virtual datasets.

In order to avoid recreating data each time it is requested, we assume existence of a dataset replica catalog (DRC) that records the concrete replica or replicas associated with each virtual dataset.

Job Catalog

We also demand complete provenance tracking, i.e. the complete history of all concrete datasets, not just the essential provenance recorded in the VDC. We define any action that creates a dataset to be a job and record the history associated with each job in a job catalog. We assume the existence of a job object describing each job (implying a job type and subtypes) and an associated job repository to make these descriptions persistent. In order to facilitate selection of jobs, there will likely also be a job selection catalog. The job repository and selection catalog (if existing) constitute the job catalog.

Dataset properties

Datasets and their properties are described in a separate document, “Datasets for the Grid” [2]. Important properties for the discussion here are identity and mutability. The first guarantees that a unique identifier is assigned to any cataloged dataset. This identifier in conjunction with the catalog services, most notably the dataset repository, enables the dataset user to discover properties of a dataset. Mutability indicates whether a dataset is closed, i.e. will never change, or is open, i.e. data may be added. The possibility that data may be removed as well as added is not considered here to simplify the discussion.

Use cases

We begin with some use cases to identify the required datasets and catalogs. Other usage patterns can be imagined. Specific choices allow us to outline a definite model that is intended to provide an initial implementation. The lessons learned from that model will be used to refine the use cases and the catalogs.

Data acquisition

The primary view presented by the VDC is one of transformations: a dataset is produced by applying a transformation (application plus task) to an existing dataset (the input or parent dataset). However, there must be a starting point for this chain of datasets. For real data (as opposed to that produced in simulation), that starting point is the data acquisition system associated with the detector. We assume that data acquisition takes place in a

series of non-overlapping runs. For each run, detector conditions are established, a set of triggers is defined and these remain stable while data is acquired for some period of time.

UC1.1 Begin a new run

At the beginning of a new run, the production manager defines an empty virtual dataset that will reference all the raw data produced during that run. This raw run dataset is entered into the VDC with no parent dataset or with its Monte Carlo parentage. The dataset is entered as an index in the DRC.

At this point and until the end of the run, the virtual raw run dataset is open. A user may find more events in the dataset as time passes.

The run dataset is entered in the DSC (dataset section catalog) and some of the attributes (such as start time and mutability) are assigned non-default values.

Accessed catalogs: VDC, DRC, DSC

UC1.2 Acquiring data

Raw data is acquired and written into files such that the data for any one event is contained in a single file and each file only contains data for the current run. When a file is complete, it is used to construct a raw single-file dataset which is recorded in the SFDC (single-file dataset catalog). This catalog is used to guarantee only one single-file dataset is defined for any given file. A logical (rather than physical) file name or ID is recorded.

At regular intervals, the raw single-file datasets for the current run period are used to construct a new closed compound dataset. The replica for the run in the DRC is updated to reference this new dataset. The event count and event list in the (open) virtual run dataset is updated accordingly.

The single-file and intermediate concrete run datasets are not entered in the VDC. The finest granularity there is the run.

Accessed catalogs: SFDC, DRC

UC1.3 End of a run

When a run ends, any open files are closed and added to the run dataset as described in the previous use case. The virtual run dataset is closed.

DSC attributes for the run dataset are assigned values. These attributes include mutability (closed), detector conditions, trigger settings, instantaneous and integrated luminosity, and perceived quality.

Accessed catalogs: DSC

UC1.4 Checking a raw data file

Any time during or after a run, the current shifter may submit a job to validate the data in a file. The shifter provides the transformation (application plus task) to carry out the validation. The input dataset is the single-file dataset corresponding to the file of interest. The output of the job is another dataset, typically consisting of histograms to be examined by the shifter.

The association between application, task, and input and output datasets is recorded in the job catalog. Of course, the definitions of all of these are recorded in repositories.

Accessed catalogs: SFDC, job

UC1.5 Checking a collection of files

The shifter may prefer to check all the files for a particular shift or day. In this case, the shifter merges the appropriate single-file datasets to create a new compound dataset and then submits a job with the latter as input.

Accessed catalogs: SFDC, job

Reconstruction

Most of the data coming off the detector is reconstructed in the same way. The raw data is used to create ESD (events summary data) which is used to create AOD which is then used to create tag data. The ESD and AOD are stored as event data objects accessible through event headers which are accessed from POOL event collections. The tag data is stored as event attributes associated with the entries in these collections. We assume that a collection of monitoring histograms is created in conjunction with the ESD and AOD.

The ESD, AOD and tag may be produced as part of a single job or may be produced in separate jobs. For definiteness, we assume separate jobs in our use cases. In any case it is useful to have separate entries in the VDC so that later steps can be redone using new transformation version without having to redo the entire sequence. Use cases for AOD and tag that are similar to those for ESD are omitted.

UC2.1 Virtual reconstruction of a run

The first step in data processing is to reconstruct the raw data. After the virtual raw dataset for a run is entered in the VDC, a virtual reconstructed dataset may also be added to that catalog. The entry includes the raw dataset as the input dataset and the current default reconstruction values for the application and task.

The content of the new dataset includes the raw data from the input dataset and the produced ESD and monitoring histograms.

Accessed catalogs: VDC

UC2.2 Concrete reconstruction of a completed run

After the data for a run are acquired and the raw dataset is closed, the production manager triggers a job to carry out concrete reconstruction, i.e. the creation of a replica corresponding to the virtual reconstructed dataset for the run. The entry in the VDC provides the application and task to use for this job.

The processing system consults the DRC to find the corresponding concrete compound dataset. The dataset is split into sub-datasets which are processed in separate jobs. The compound nature of the dataset aids in splitting along file boundaries. Each job produces a reconstructed dataset and these are merged to form the concrete reconstructed dataset

for the run. The DRC is updated to register this concrete dataset as a replica of the virtual reconstructed dataset for the run.

The processing system is smart enough to recognize those parts of the dataset that have already been reconstructed with the same algorithm plus task and to use the corresponding output datasets instead of producing them again.

During the processing, the incomplete reconstructed dataset merging datasets from the so-far completed jobs is available for examination. The dataset and its constituents are open if the job has not completed successfully.

Accessed catalogs: VDC, DRC, job

UC2.3 Reconstruction of a file

The production manager may elect to reconstruct individual raw data files as they produced rather than waiting for the end of the run. This is accomplished by submitting a job with the corresponding single-file dataset and the same application plus task used to define reconstruction in the VDC. The input dataset might be a collection of single-file datasets if that is a more natural processing unit.

The job is recorded in the job catalog and the result used to avoid repeating the processing when the concrete run dataset is created.

Accessed catalogs: VDC, job

UC2.4 Virtual production of AOD

The next step in production is to use the ESD to produce AOD (analysis-oriented data). A virtual AOD dataset for the run is entered in the VDC with the ESD dataset for that run as the input dataset and the current default AOD values used for the application and task. The application and task advertise that they only need the ESD part of the input dataset so that a processing system can know that the raw and histogram pieces are not needed.

The content of the output dataset is the input raw, ESD and ESD histograms and the produced AOD and AOD histograms.

Accessed catalogs: VDC

Combining runs

The division of data into runs is largely an artifact of acquisition and processing. Physics analyses will make use of data that spans many runs and often many years. Some runs or parts of runs will be discarded when problems in the detector are discovered later. For simplicity, we assume that the good data for a run period may be defined by taking all the data for a collection of runs. This combined dataset is the starting point for physics analysis.

UC3.1 Combined reconstructed dataset

The collaboration decides on a run period to be used as the basis for physics results, e.g. all good runs before the December shutdown. The production manager creates a virtual dataset that merges the virtual tag datasets for all good runs and enters the new dataset

with parents in the VDC. By definition, the new dataset includes raw, ESD, AOD and tag data as well as monitoring histograms.

An entry for the new dataset is created in the DSC with appropriate metadata.

Accessed catalogs: VDC, DSC

Event selection

A combined dataset is typically very large including something like 10^9 events per year of data acquisition. Typically the first stage of analysis is to create filtered datasets with much smaller numbers of events by defining transformations that make selections based on event characteristics. These transformations are applied to the combined dataset or other filtered datasets. Ideally, the tag carries sufficient data to make these selections but it is also possible to make use of the AOD or ESD.

UC4.1 Virtual event selection

A selection is defined by creating a selection transformation (application plus task) and then adding a VDC entry with this transformation and the combined dataset or another selected dataset as input. The virtual output dataset has the same event content (the histograms are dropped) but only the selected events. A DSC entry is created for the new dataset including information about the chain of selections from the starting combined dataset. This catalog may also contain a flag indicating that this is a dataset suitable for physics analysis.

Accessed catalogs: VDC, DSC

UC4.2 Concrete event selection

The first request for a concrete replica of the previous output dataset (or some other action) triggers the processing system to create a job that fetches a concrete replica of the input dataset and then splits, processes and merges to create the concrete output dataset. The association between output virtual and concrete datasets is then recorded in the DRC. The event count and list for the virtual dataset can be filled in at this point.

Accessed catalogs: DRC, job

UC4.3 Separation into streams

The event data in the concrete combined dataset may be organized into streams, i.e. have physical placement based on event characteristics. If this is the case, then it is natural to define selections that reflect this placement.

UC4.4 Copying event data

Most filtered datasets will contain only a tiny fraction of the events in the combined dataset. A concrete representation of the filtered dataset that uses the same files as the combined dataset still holds the same volume of data. A significant reduction in this data volume may be made by eliminating those files that hold no selected events. However, most of the data volume is still rejected events that happen to reside in the same file as a

selected event. Another significant reduction in volume may be made by copying only the selected events into new files.

A physics coordinator or user defines a job that takes a virtual selected dataset as input and copies the selected events to new files. The new concrete dataset is registered in the DRC as another replica for the same virtual dataset.

The processing system might also do this automatically, i.e. copy selected events before transferring data if only a small fraction of events in the files are required.

Accessed catalogs: DRC

Analysis

A physicist performing analysis will submit jobs starting from filtered datasets. These jobs may carry out further selections or may run algorithms that produce more event data or fill analysis objects such as histograms or ntuples. The former typically operate on the tag and the latter on the AOD but there will be circumstances where the physicist needs to access other parts of the event data.

UC5.1 Analysis job

A physicist defines a transformation (application plus task) and uses the DSC to select a virtual filtered dataset. These are submitted to an analysis service. The analysis service finds a concrete replica for the input dataset and then applies the transformation to that dataset to create the output dataset (new data and/or analysis objects). The job submission may also include configuration parameters where the user specifies whether the new dataset is to be recorded in the VDC and DRC. If so, the analysis service updates these catalogs.

Accessed catalogs: DSC, DRC, VDC

UC5.2 Select a task

A physicist submitting an analysis job will typically find an existing task and modify it for the new job. It is natural to assume that the task is found in a task selection catalog (TSC) and that new tasks are added to the catalog either automatically or by their creator.

Accessed catalogs: TSC

UC5.3 Select a dataset

As already indicated, physicists will expect to start an analysis from an existing dataset. They will make this selection based on metadata which summarizes the selections or intent of the selections rather than the detailed provenance chain. Thus we expect the physicist to consult the DSC to find the starting dataset.

Accessed catalogs: DSC

Required catalogs

From the above discussion and use cases, we infer the following catalogs are required:

Application repository
Task repository
Task selection catalog (TSC)
Dataset repository
Dataset replica catalog (DRC)
Dataset selection catalog (DSC)
Single-file dataset catalog (SFDC)
Virtual data catalog (VDC)
Job catalog (repository and possibly selection catalog)

Conclusions

Use cases for production and analysis have been presented using AJDL and a virtual data model. The required catalogs have been identified.

References

1. "AJDL: Analysis Job Description Language", D. Adams, http://www.usatlas.bnl.gov/ADA/docs.
2. "Datasets for the Grid", D. Adams, http://www.usatlas.bnl.gov/~dladams/dataset.